scrape_imdb_top250

July 8, 2023

[ ]:
```
import numpy as np
import polars as pl
from selenium import webdriver
from bs4 import BeautifulSoup
```

[ ]:
```
# Top 250 Movie URL
url = 'https://www.imdb.com/chart/top/'
```

[ ]:
```
# Launch the Chrome browser
driver = webdriver.Chrome()
driver.get(url)

# Parse the HTML using BeautifulSoup
soup = BeautifulSoup(driver.page_source, 'html5lib')

# Close the browser
driver.quit()
```

[ ]:
```
# Find all movie div elements
movies = soup.find_all('div', {'class': 'sc-14dd939d-0 fBusXE cli-children'})
rows = []

for movie in movies:
    # New row
    row = []

    # Title
    title_raw = movie.find('h3', {'class': 'ipc-title__text'})
    title = ' '.join(title_raw.text.split()[1:])
    row.append(title)

    # Rate
    rate_raw = movie.find('span', {'class': 'ipc-rating-star␣
 ↪ipc-rating-star--base ipc-rating-star--imdb ratingGroup--imdb-rating'})
    rate = rate_raw['aria-label'].split()[-1]
    row.append(float(rate))
```

```
    # Year
    year_duration_raw = movie.find_all('span', {'class': 'sc-14dd939d-6 kHVqMR␣
 ↪cli-title-metadata-item'})
    year = year_duration_raw[0].text
    row.append(int(year))

    # Duration
    duration_raw = year_duration_raw[1].text
    duration_splitted = duration_raw.split()
    hour = duration_splitted[0][:-1]
    if len(duration_splitted) > 1:
        minutes = duration_splitted[1][:-1]
    else:
        minutes = 0
    duration = int(hour) * 60 + int(minutes)
    row.append(duration)

    # Append row to rows
    rows.append(row)
```

[ ]:
```
# WITH POLARS
# Create a Polars DataFrame
df = pl.DataFrame(rows, schema=[("title", pl.Utf8), ("rate", pl.Float32),␣
 ↪("year", pl.Int32), ("duration", pl.Int32)])
rank = np.arange(1, 251, dtype=np.int32)
df = df.with_columns(pl.lit(rank).alias("rank"))

# Write the DataFrame to a CSV file
df.write_csv("top250imdb_2023-07-07.csv", separator=",")
```

[ ]:
```
# WITH PANDAS
# import pandas as pd
# columns = ['title','rate','year', 'duration']
# df = pd.DataFrame(rows, columns=columns)
# df = df.reset_index()
# df["index"] = df["index"] + 1
# df.rename(columns={"index": "rank"}, inplace=True)
# df.to_csv("top250imdb_2023-07-07.csv", index=False)
```